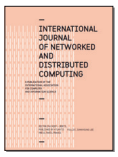
International Journal of Networked and Distributed Computing
Vol. **8**(**3**); June (2020), *pp.* 139–151
DOI: https://doi.org/10.2991/ijndc.k.200515.002; ISSN 2211-7938; eISSN 2211-7946
https://www.atlantis-press.com/journals/ijndc



Research Article

Performance Analysis of Blockchain-based Access Control Model for Personal Health Record System with Architectural Modelling and Simulation

Thein Than Thwin, Sangsuree Vasupongayya*

Department of Computer Engineering, Faculty of Engineering, Prince of Songkla University, Hatyai, Songkhla 90110, Thailand

ARTICLE INFO

Article History

Received 11 December 2019
Accepted 10 February 2020

Keywords

Security
cryptography
palladio model
health records

ABSTRACT

The Personal Health Record (PHR) could be seen as a preventive care solution to the incoming aging society. The blockchain-based PHR system has been proposed recently to enhance the security and privacy for the PHR data. Consequently, the performance becomes a concern for blockchain-PHR integration because of the blockchain performance issues in the past. Thus, this article presents the performance analysis of the blockchain-based PHR system to ensure the usability in practice. The proposed blockchain-based PHR system prototype is implemented and the architectural model for the blockchain-based PHR system is also constructed. The key parameters for the architectural model are extracted from the prototype. Experiments are conducted with various data sizes including 128, 512 KB, 2, 8 and 32 MB. The result shows that storing 32 MB of the PHR data takes 4.84 s and retrieving the same PHR data takes 5.19 s. The result of simulating the architectural model shows that the proposed blockchain-based PHR system can response within 4 min for 60,000 accesses each day. The performance results indicated that the proposed blockchain-based PHR system can work within the emergency response time of 8 min and it is usable with an efficient computational cost. Further evaluation on the distributed design of the proposed blockchain-based PHR system is planned for our future work.

1. INTRODUCTION

The privacy and trust of Personal Health Record (PHR) system [1,2] are improved by applying the blockchain technology, in our prior works [3]. The issues that affect the use of blockchain in the PHR development were introduced and addressed with an existing private blockchain and cryptographic mechanisms in our prior work [4]. However, the usability evaluation is still lacking in our previous work [4]. Only the achievement of the privacy and the security as well as the effectiveness of the proposed blockchain-based PHR system was shown. Since, the performance is a concern for the use of blockchain for the PHR development, the usability of the proposed blockchain-based PHR system must be addressed. For that reason, the performance analysis is conducted in this work.

As a result, an architectural model will be used for predicting the performance properties [5]. Thus, a model-driven approach for predicting the performance of the blockchain-based PHR system is constructed. This article extends our prior work [4] by evaluating the proposed blockchain-based PHR system on various number of users and various PHR data sizes under a real world scenario. An architectural model is proposed to analyze the performance of the proposed blockchain-based PHR system, with the key aspect of ensuring the usability in practice.

*Corresponding author. Email: vsangsur@coe.psu.ac.th

To characterize the key components of our prior blockchain-based PHR system, a prototype system is firstly implemented and the execution time of each component is extracted. Then, an architectural model is proposed by using the execution times extracted from the prototype system for each associated component. In developing the architectural model, Palladio workbench [6] which is an architectural modelling tool is used. Generally, low performance is mainly the effect of inappropriate architectures rather than that of weak implementation [7]. Thus, the aim of this work is to evaluate our proposed blockchain-based PHR system in its early lifecycle stage.

This paper is structured as follows. In Section 2, some related works and the outline for our blockchain-based PHR system which is proposed in prior work [3,4], are presented. Section 3 discusses how the required parameters for an architectural model simulation are benchmarked. The Palladio Component Model [6] for analyzing the performance of our proposed blockchain-based personal health record systems is also described. The results of the prototype implementation and the system-level analysis from the model are evaluated and discussed in Section 4. Finally, the paper is concluded in Section 5.

2. BACKGROUND AND RELATED WORKS

Personal health record and other digital health record systems have the potential to improve health outcomes, support care

coordination, and improve communication [8]. Whereas these digital health record systems have the potential for better health care, their security issues must be concerned [9]. Blockchain technology [10] presents numerous opportunities for development of such digital health record systems. Many researches used cryptographic techniques and blockchain for privacy and security of digital health record systems. Majority of existing research in health record field used attribute-based encryption scheme and the semi-trusted servers to store the health related data in privacy preserving manner [11–19]. These researchers tried to create the access control by directly encrypting the real health data. As a consequence, the performance might be a concern in reality because the attribute-based encryption could cause a growing computational cost linearly with the number of unrevoked users. In Azaria et al. [20], linn and Koo [21] and Ivan [22], the blockchain-based data sharing systems were suggested by adding the blockchain based access control layer to existing databases of the providers. However, only the idea was proposed and the usability of the system was proved from the security point of view. Wang and Song [23] used attribute-based encryption to encrypt the EHR data for a fine-grained access control and the identity-based signing is used for implementing the digital signatures. They also used blockchain technology to ensure a tamper resistant property for medical data and traceability. A demonstrating application was shown for a medical insurance scene. Again Li et al. [24] tried to break the medical data into pieces and stored on multiple blocks. Li et al. [24] evaluated the processing time for different sizes of data. Roehrs et al. [25] tried to collect the EHR data into PHR system by using a blockchain idea to support the interoperability. And then, the performance of the model proposed in Roehrs et al. [25] was evaluated in Roehrs et al. [26]. The authors proposed a prototype system and tested with the some data.

In our previous works [3,4] the blockchain based PHR system was proposed and the achievement of our proposed system was also demonstrated from the security point of view. Thus, in this work, the architectural model will be used for analyzing our previously proposed blockchain-based PHR system. The Palladio workbench [6] is used as a modelling tool for our architectural model. Moreover, Palladio workbench supports a "UML-like" interface for model construction. Palladio workbench is flexible for extensions such as an architectural optimization [27] and the new qualities [28]. The accuracy of performance models of traditional cloud and database based systems has been previously studied in de Gooijer et al. [27] and Brunnert et al. [29]. Brunnert et al. [5] suggested that the performance analysis can be performed with analytical solvers or simulation engines. Thus, the simulation engine will be used in this work.

2.1. Blockchain-based PHR System

To support the development of the architectural model, a prototype system is implemented according to our prior blockchain-based PHR system [4]. Thus, the previously proposed blockchain-based PHR system is presented in this section. The system design of the previously proposed blockchain-based PHR system is shown in Figure 1.

As shown in Figure 1, the primary use case of the system is that the PHR owner can share their PHR data with others. Thus, the PHR system allows the PHR owner to upload the PHR data. In addition, the users can also download the PHR data. As a result, the system design includes two main operations (storing the PHR data and retrieving the PHR data). The detailed workflows of these two operations are illustrated in Figures 2 and 3 respectively. The execution times of the components that support these two operations will be analyzed in this work.

According to the design shown in Figure 1, there exist five elements in the proposed blockchain-based PHR system including the PHR-owner client, the user client, the gateway server, the cloud storage server and the blockchain server. However, there are only four elements when a storing operation is performed. The four elements include the PHR owner client, the gateway server, the cloud storage server and the blockchain server, as shown in Figure 2. To store the PHR data, the PHR owner client performs the following processes:

- Calculate the hash code of PHR data.
- Encrypts the PHR data with the owner public key.
- Creates the digital signature.
- Creates the re-encryption keys for permitted users.
- Sends the resulted data elements to the gateway server.

Thus, hashing time, encryption time, re-encrypt key generation time, signing time and data sending time in PHR owner client must be analyzed for the storing operation. The gateway server performs the following processes:

- Verifies the signature of the PHR owner.
- Stores the encrypted data on the cloud storage.
- Stores the data id, link and access list locally.
- Creates the server signature.
- Stores the meta-data on the private blockchain.

Thus, the owner signature verifying time, the data uploading time, the local data storing time, the server signing time and the blockchain accessing time of the gateway server must be evaluated for a storing operation.

For the retrieving operation, the user client, the gateway server, the cloud storage server and the blockchain server are interactively working together as shown in Figure 3. To retrieve the PHR data, the user client performs the following processes:

- Searches the PHR data via the blockchain.
- Verifies the owner signature.
- Verifies the server signature.
- Creates the digital signature.
- Send the request to the gateway server.
- Decrypt the received PHR data.

Thus, the time for searching the PHR data on the blockchain, the owner signature verifying time, the server signature verifying time, the user signing time, the request sending time and the decryption time of the user client must be evaluated. The gateway server performs the following processes:

- Verify the signature of the user.
- Stores the request log on the blockchain.

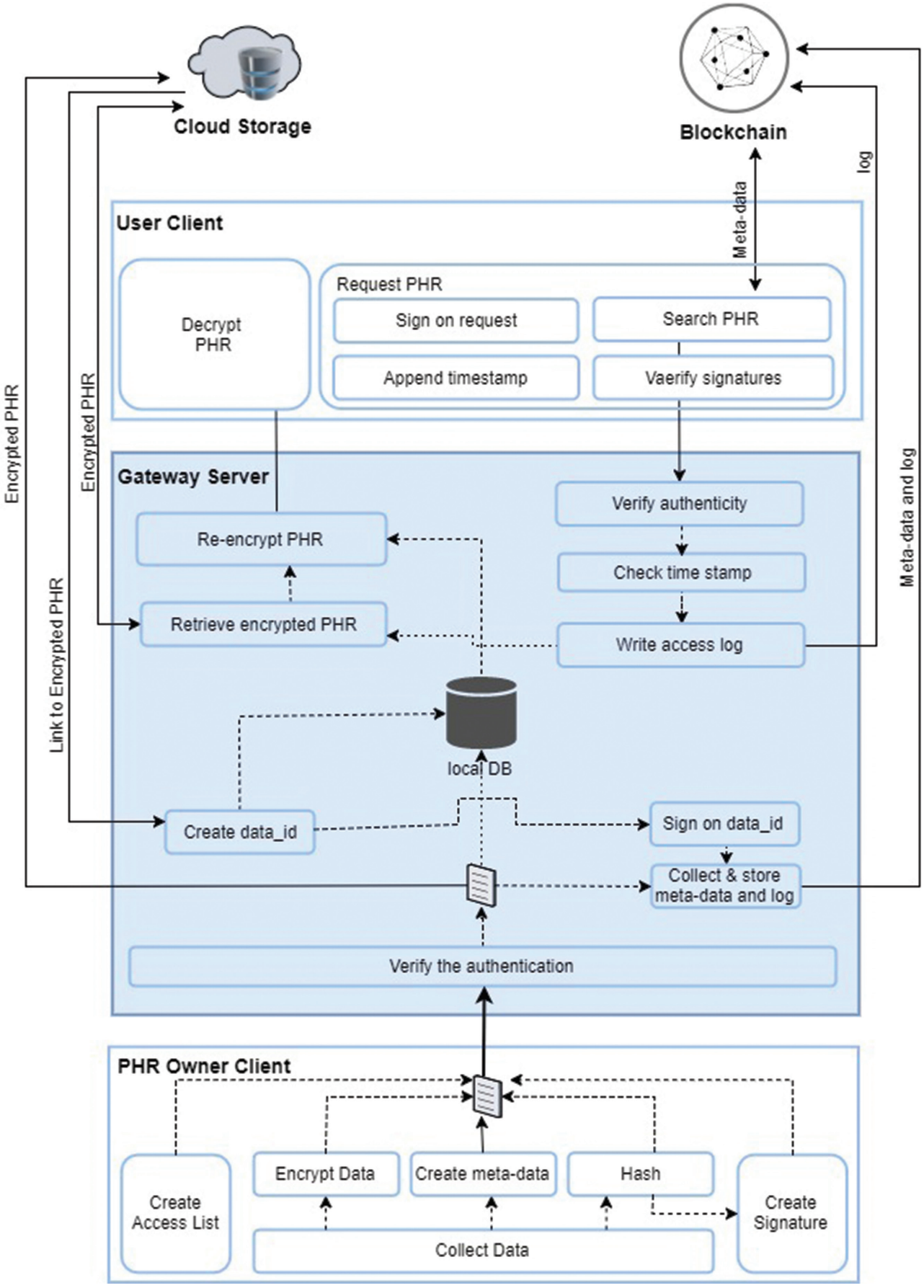


Figure 1 | The system design of blockchain-based PHR system.

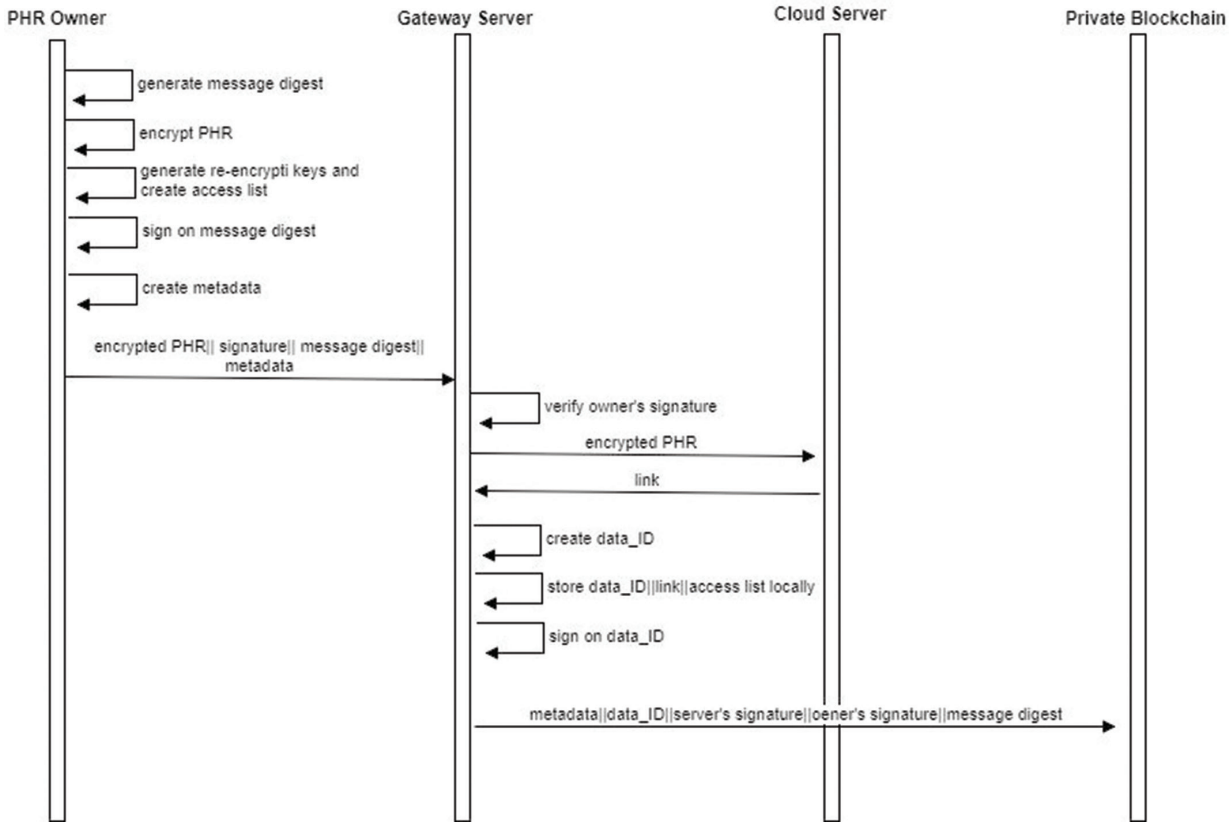


Figure 2 | Workflow for storing a PHR.

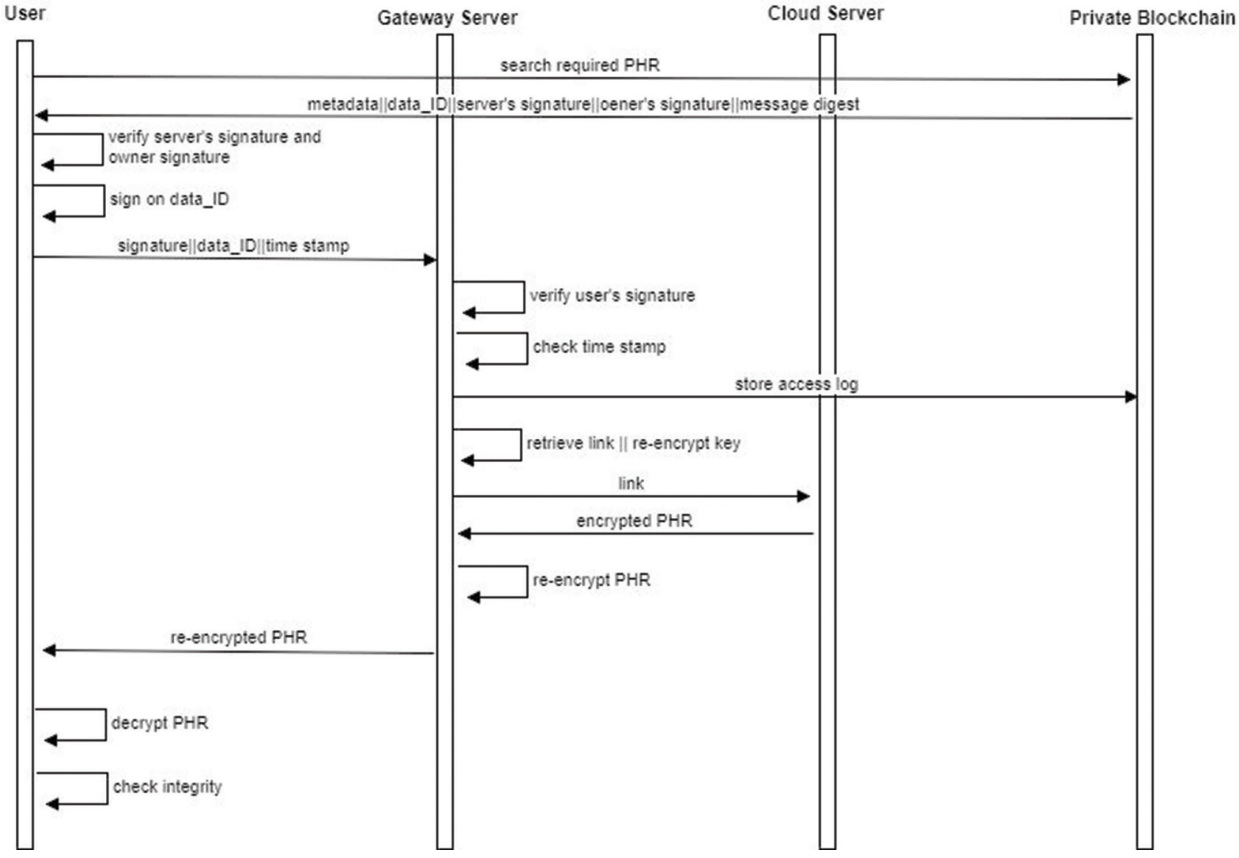


Figure 3 | Workflow for retrieving a PHR.

- Retrieves the encrypted data from the cloud storage.
- Re-encrypts the PHR data.

Thus, the user signature verifying time, the time for saving the log data on the blockchain, the data downloading time and the re-encryption time must be evaluated.

3. METHODOLOGY

The blockchain-based PHR system prototype, proposed in our previous work [4], is firstly constructed. Then the architectural model of the system is then constructed for analyzing the performance.

3.1. Benchmarking the Processing Time of Components

The most important key parameter for our architectural model is the execution time of each component of the model. To evaluate the execution time, a prototype version of our proposed blockchain-based PHR system is constructed. The prototype system is developed by creating five virtual machines on VMware vSphere 6 Hypervisor. The host machine is with IntelR Core TM i7-3770 CPU @ 3.4 GHz processor 32 GB RAM and 4.1 TB HDD. The communications among virtual machines are done by the virtual switch of ESXi for increasing the network speed and reducing the network latency. The prototype of the PHR owner client, the user client, the gateway server, the cloud storage server and the blockchain server are implemented on each of the five virtual machines.

The Proxy Re-encryption Scheme (PRES) which is an asymmetric crypto system that enables its users to share their decryption capabilities to others [30] is implemented using the AFGH algorithm [31] and Advanced Encryption Standard (AES) [32]. Under the PRES, the data is encrypted with a symmetric cipher AES and the symmetric key is also encrypted to control the access. So, the ciphertext is the combination of the encrypted data and the encrypted symmetric key. The AFGH algorithm is performed on the symmetric key for the encryption and decryption processes. The Hyperledger blockchain network is created in Docker environment [33] with node.js [34]. The blockchain network simply contains the end-user node, the orderer node and the two peer nodes. The Hyperledger fabric 1.0 version is used and the dummy consensus is used to reduce the cost.

3.2. Synthetic PHR Workloads

The synthetic PHR data of several data sizes are used for benchmarking in this work because the real PHR date is difficult to achieve. Variety of the PHR related data sources are studied to generate the synthetic PHR data. There is no standard for the PHR workloads.

A synthetic PHR data which can mimic the real PHR data are proposed for evaluating their works in Tang et al. [1]. Most PHR data are in the form of a document file such as Continuity of Care Document (CCD) file, continuity of care record file, and a document file type. In addition, the PHR data can exist in a form of a media file such as an image file (Magnetic Resonance Imaging (MRI), X-ray or electrocardiogram (ECG) image file), an audio file (a doctor visit conversation) and a video file (an operation video or a healthcare instruction video). Various document and media file sizes and types are collected and used as a synthetic workload in Tang et al. [1], Sobhy et al. [35] and Wangthammang and Vasupongayya [36]. Their workloads include an MRI file (a 20 KB jpeg file) which is collected from Lijun et al. [37], a CCD file (a 27 KB xml file) which is collected from Li et al. [38], a patient information file (a 30 KB xlsx file) which is created as a representative of several PHR data types, a heartbeat sound file (a 154 KB ogg file) which is collected from Bahga and Madisetti [39], an ECG picture file (a 393 KB jpg file) which is collected from Li et al. [40], a patient information file (a 431 KB docx file) which is created as a representative of several PHR data type, the CCD files (a 617 KB xml file and a 679 KB pdf file) which is collected from Dong et al. [41], an X-ray file (a 4 MB png file) which is collected from Bahga and Madisetti [39], an audio file (a 8 65 MB mp3 file) which is created as a representative of a voice conversation file and a video file (a 27 MB mp4 file) which is collected form YouTube [42]. The PHR template which is created as a plaintext file supported by VA Personal Health Record Sample Data - Data.gov [43] is 169 KB in size. Majority of the medical video clips provided by MedCram [44] are <30 MB in size.

According to the result of our study, most of the largest files in PHR workloads are video files. The video clip which is encoded in MP4 format with the length of 19:52, frame resolution of 1280 × 720 pixels, the frame rate of 29 frames per second, two stereo audio channels and the bit rate of 183 KB per second is 26.6 MB in size. Thus, we use 128 KB synthetic data as the smallest size and increase the size by multiplying by 4 in order to create our synthetic PHR data up to 128 MB to cover each size of the PHR data. Moreover, there are many free applications and open-source projects which can split or merge video files such as MP4Tools [45], Machete Video Editor [46], Format Factory [47], Avidemux [48] and Free maker Video Converter [49]. As a result, in our synthetic PHR workload, the PHR data files are created in sizes of 128, 512 KB, 2, 8, 32 and 128 MB.

3.3. Constructing the Palladio Component Model for Performance Analysis

To simulate an architectural model in the Palladio bench, the following sub-models are required.

- Repository Model which specifies a set of components that can later be deployed within the system.
- System Model which is a concrete component-based software system, created by using the available components in the component repositories (Repository Model).
- Environment Model which specifies the CPUs, the hard disk drives, the networks, and the resources for each server or node in the running environment.
- Allocation Model which allocates the components assembled within the system to the resource environment.
- Usage Model which specifies the behaviors of the users.

To simulate the proposed blockchain-based PHR system, the components that will execute the corresponding process are created in the repository model of the Palladio Component Model (PCM). The resulted execution times from the experiment of the prototype

Table 1 | The core processes of the proposed solution

No.	Process	Execution
1	Sign verify	0.1
2	Data upload	DoublePDF [(157.4;1/6) (221.6;1/6) (273.6;1/6) (457.5;1/6) (654.2;1/6) (2150.87;1/6)]
3	Locally store	38.4
4	Sign	1.6
5	Blockchain access	3398.4
6	Re-encrypt	DoublePDF [(30.6;1/6) (31.5;1/6) (34.3;1/6) (58.8;1/6) (79.7;1/6) (58.8;1/6) (79.7;1/6)]
7	Data download	DoublePDF [(38.0;1/6) (78.3;1/6)(152.3;1/6) (214.8;1/6) (469.3;1/6) (1093.034;1/6)]

system are configured into the corresponding components of the model. However, the architecture of our proposed solution contains five elements. The PHR owner and the user access the PHR system via their own clients. When the number of users increases, the number of clients will also increase accordingly. The owner client and the user client are not included in the architecture model. The gateway server, the cloud storage server and the blockchain server are modelled. The core processes and their execution times in the PCM are shown in Table 1. The detail of these values will be discussed in Section 4.

3.3.1. Constructing the repository model

The proposed PCM repository model contains the components and interfaces that are required to construct our proposed blockchain-based PHR system in real world, as shown in Figure 4.

Each component needs to be specified with the Service Effect Specification (SEFF). SEFF is an abstraction of the component behavior embedded in the component model. SEFFs refer to the method signatures and the parameters that are declared in the

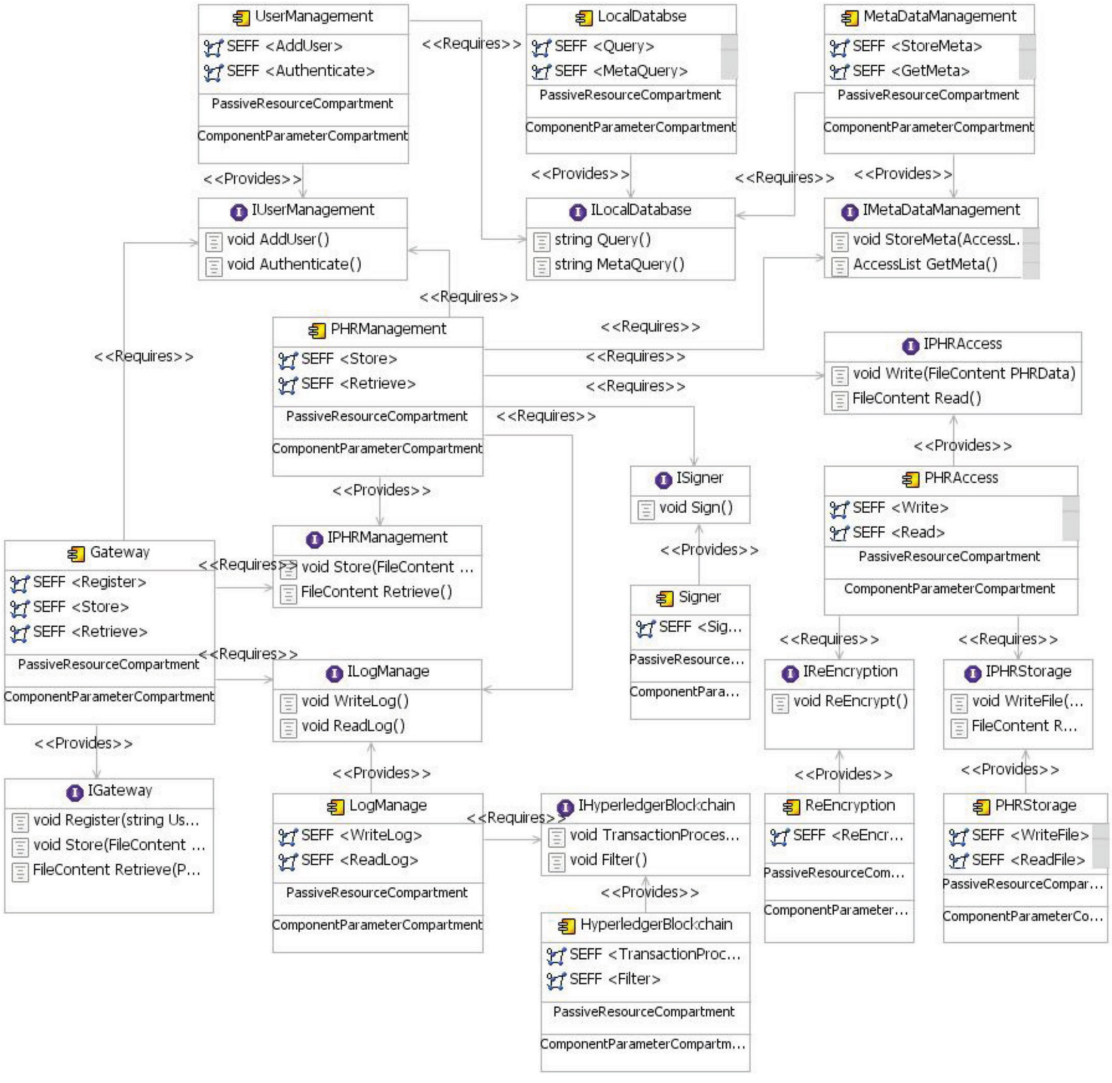


Figure 4 | PCM repository model diagram.

interfaces. The control flow between calls to each required service, the parametric dependencies, and the resource usage must be added. Resource requirement such as execution times can be configured in the SEFF. Thus, the execution times shown in Table 1 are added to the corresponding SEFF of each component in our repository model to reflect the components of our prototype system.

3.3.2. Constructing the system model

The system model characterizes the component assembly of our proposed blockchain-based PHR system. The system model supports the inter-component structure of our proposed blockchain-based PHR system by assembling the components which are defined in our PCM repository model. The system model can be used for estimating the performance of our proposed blockchain-based PHR system for different scenarios. However, in this work, the system model is specified only as shown in Figure 5 because, the design alternative is not addressed in this work.

3.3.3. Constructing the execution environment model

The execution environment model defines the hardware nodes and the network in the running environment of our proposed blockchain-based PHR system. For the resource environment model, the gateway server, the cloud storage server and the blockchain server are constructed to mimic the test-bed as shown in Figure 6.

In this work, the performance of the proposed blockchain-based PHR system is analysed using the CPU that is defined with the parameter for processing rate 1000 × 1000 cycles per second and the network is defined with latency 0.3493513513514 and throughput as 1000 × 1000. The latency is defined by testing the communication between two virtual machines in our test-bed with ping test and used the average value.

3.3.4. Constructing the component allocation model

The component allocation model describes how the components of our proposed blockchain-based PHR system are deployed on the hardware nodes. It defines which component is executed on which part of the execution environment. The real encrypted PHR storage and the blockchain component are allocated on individual machines, i.e., the cloud storage server and the blockchain server, and all other components are allocated on the gateway server as shown in Figure 7.

In the future work, the machine in the environment model can be added and the allocation model can be changed for any design alternative.

3.3.5. Constructing the usage model

The usage model defines how the users interact with our proposed blockchain-based PHR system. In our architectural model, the workload is an open workload that is specified by the inter-arrival times of requests. Thus, the large numbers of users can be specified with the arrival rate by assuming the arrival time by each distribution. The primary use case of our proposed blockchain-based PHR system is that the PHR owners can share their PHR data with others. Thus, the proposed blockchain-based PHR system allows

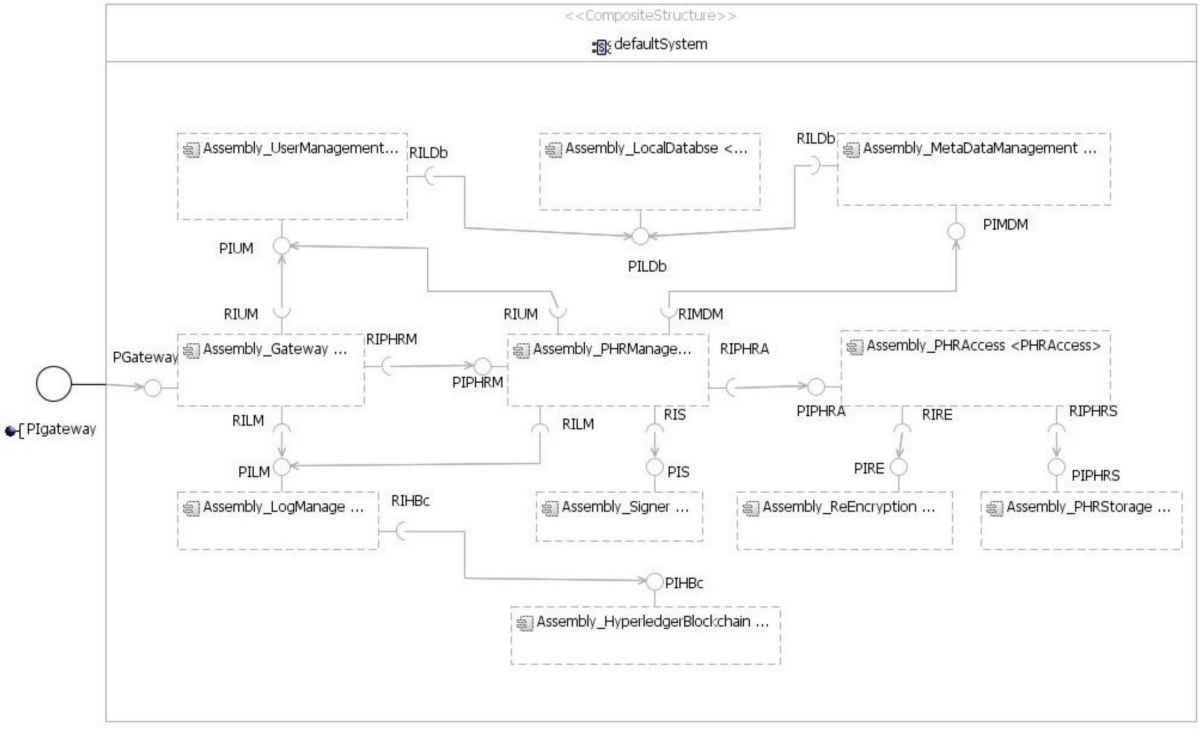


Figure 5 | PCM system model diagram.

Figure 6 | PCM execution environment model diagram.

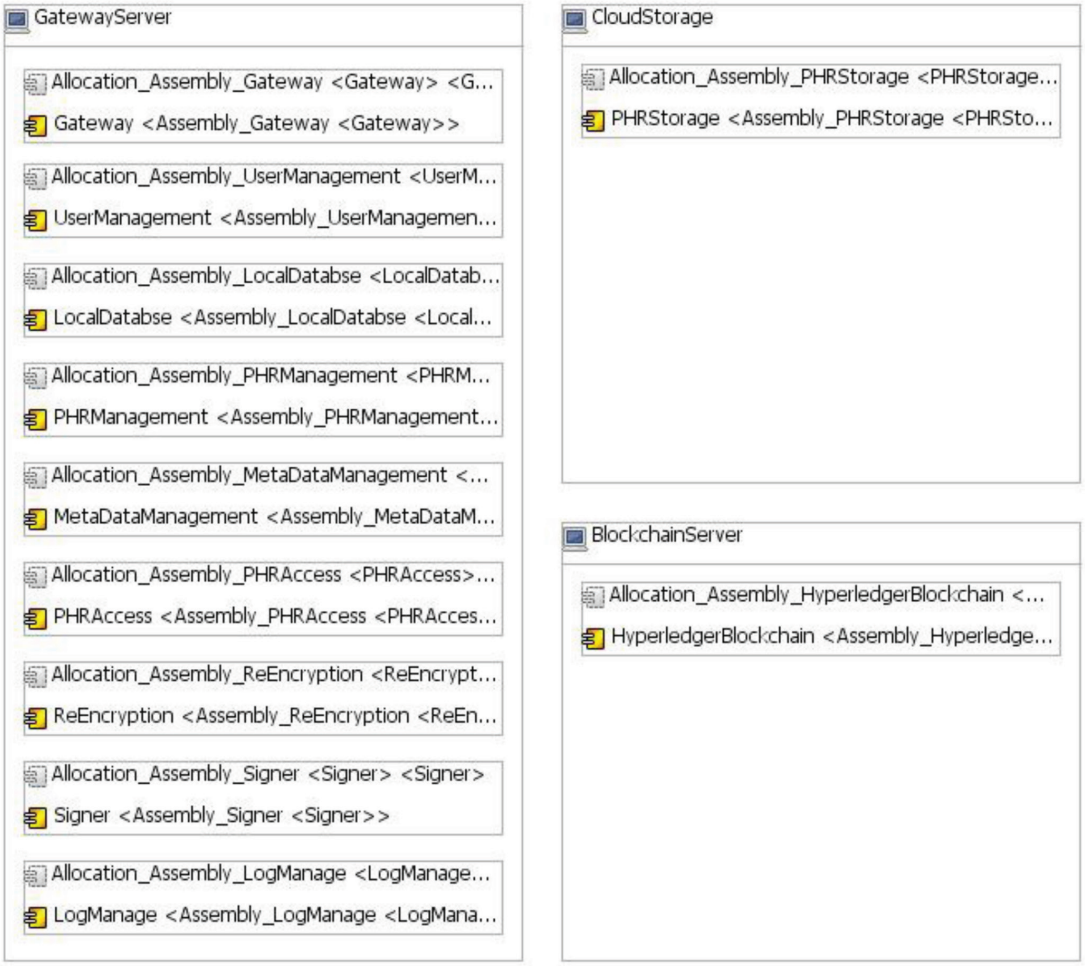


Figure 7 | PCM component allocation model diagram.

the registered PHR owners to upload their PHR files. In addition, the registered users can also download the PHR files according to their permissions. Thus, the scenario for our architectural model is constructed based on the workflows shown in Figures 2 and 3. The usage model diagram is shown in Figure 8.

4. RESULTS AND DISCUSSION

The results of the prototype system and the result of simulating the architectural model are discussion in this section. The result of the prototype system reflects not only the execution time of each component in our proposed blockchain-based PHR system, but also reflects the execution time of the whole system with a single user. The time required for a store operation and a retrieve operation is calculated on the synthetic PHR workload consisting of 128, 512 KB, 2, 8, 32 and 128 MB in size. Then, the evaluation of the architectural model with a real world use case shows the usability of the proposed blockchain-based PHR system.

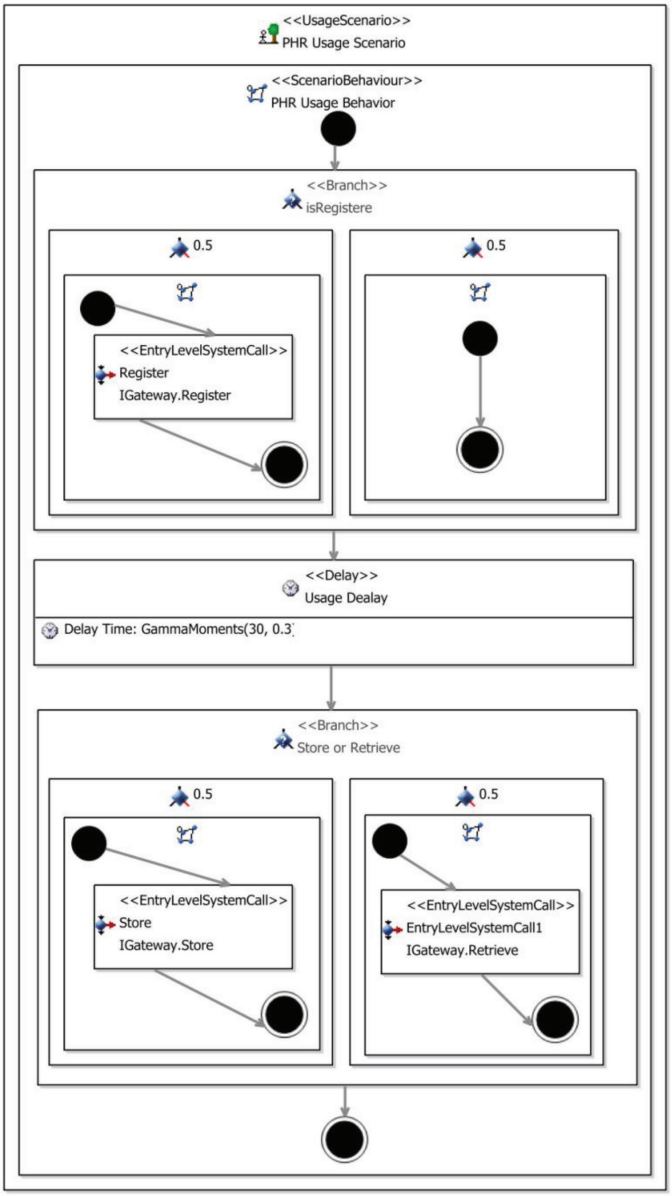


Figure 8 | PCM usage model diagram.

4.1. The Results of Running the Prototype System

To store the PHR data in the proposed blockchain-based PHR system, the PHR owner client and the gateway server operate on the store operation. The average execution time of each sub-processes of the two elements are shown in Tables 2 and 3 respectively.

Table 2 presents the detailed execution times of the PHR owner client. The PHR owner client performs five processes including hashing, encrypting, re-encryption keys generating, signing and sending the data to the gateway server in order to storing the PHR data. According to Table 2, the execution time of the three processes including hashing, encrypting and data sending, depend on the size of the PHR data while the execution time of the re-encryption key generating and the signing processes remain the same. Thus, the execution time for the three processes (i.e., hashing, encrypting and data sending) can be defined with the Probability Density Function (PDF) and the execution time for the remaining two processes (re-encryption key generating and signing) can be represented as a constant.

The gateway server also performs the five main processes including signature verifying, data uploading, storing data locally, signing and storing the log file on the blockchain. As shown in Table 3, the execution time of only one process that is the data uploading, depends on the size of the PHR data while the execution time of the remaining processes remain nearly constant. To retrieve the PHR data in the proposed blockchain-based PHR system, the user client and the gateway server operate on the retrieve operation. The average

Table 2 | The execution time of each component in PHR owner client

Data size	Hash time	Encrypt time	Re-encrypt key generation time	Sign time	Data send time
128 KB	10.29	91.18	24.16	1.16	152.73
512 KB	18.24	94.01	24.66	1.15	173.87
2 MB	40.63	101.19	26.15	1.18	268.95
8 MB	65.60	142.03	26.88	1.16	421.67
32 MB	241.80	303.79	27.00	1.31	645.70
128 MB	946.10	1828.21	27.10	1.42	2200.36

Table 3 | The execution time of each component in gateway server for storing process

Data size	Owner sign verify time	Data upload time	Local data store time	Server sign time	Blockchain time
128 KB	0.07	157.41	31.57	1.24	3372.79
512 KB	0.07	221.65	24.66	1.15	173.87
2 MB	0.07	273.65	38.40	1.30	3365.34
8 MB	0.08	457.51	33.82	1.49	3238.70
32 MB	0.06	654.280	28.62	1.60	2935.02
128 MB	0.07	2150.87	38.71	1.58	3381.05

execution time of each sub-processes of the two elements are also shown in Tables 4 and 5 respectively.

To retrieve the PHR data, the user client performs six main processes including searching on the blockchain, verifying the owner signature, verifying the server signature, signing, sending the request to the gateway server and decrypting the resulted PHR data. According to Table 4, the execution time of the decryption only depends on the data size. The gateway server will also perform four main processes including the signature verifying, saving a log file on the blockchain, data downloading and re-encrypting. The execution time of the two processes including the data downloading and the and re-encrypting depends on the data size as shown in Table 5.

According to Tables 2–5, the nearest average data (32 MB data) is used for estimating the average operation time. The time required for the PHR owner client in a store operation is 1219.606 ms and the gateway server service time of a store operation is 3619.578 ms.

Thus, the average system operation time for a store operation is approximately 4839.184 ms or 4.84 s. The time required for the user client for a retrieve operation is 1191.919 ms and the gateway server service time for a retrieve operation is 3916.822 ms. As a result, the system operation time for a retrieve operation is approximately 5108.741 ms or 5.19 s. These results show the average performance for one user. To estimate a multiple user usage scenario, a PCM model is constructed using the parameters as discussed in Section 3.3.

4.2. Validating the Architectural Model

The architectural model is simulated with the workload and the simulation result is compared with the result observed from the prototype system. When executing the prototype, the average response time for a store operation is 4.84 s while the average response time for a store operation is 5.0 s is observed from the architectural model. Thus, the simulation predicted the average response time with the relative error of 3.3% for a store operation. The average response time for a retrieve operation produced by the prototype system is 5.11 s while the average response time for a retrieve operation that is estimated by the architectural model is 5.3 s. The simulation predicts the average response time with the relative error of 3.7% for a retrieve operation. Thus, the simulation predicted the response time close to the result observed from the prototype system.

Table 4 | The execution time of each component in user client for retrieving process

Data size	Search on blockchain time	Sign verify time (owner, server)	User sign time	Request send time	Decrypt time
128 KB	785.69	0.07, 0.04	1.33	115.36	3.20
512 KB	820.48	0.07, 0.04	1.29	124.30	6.04
2 MB	751.15	0.07, 0.04	1.31	110.77	16.63
8 MB	770.61	0.07, 0.04	1.23	136.25	59.41
32 MB	823.37	0.07, 0.04	1.75	127.79	238.90
128 MB	796.67	0.07, 0.04	1.39	128.77	1814.79

Table 5 | The execution time of each component in gateway server for retrieving process

Data size	User sign verify time	Save log on blockchain time	Re-encrypt time	Data download time
128 KB	0.11	3304.62	30.59	38.02
512 KB	0.10	3288.55	31.50	78.27
2 MB	0.10	3308.91	34.28	152.31
8 MB	0.11	3398.48	58.75	214.84
32 MB	0.13	3367.66	79.70	469.33
128 MB	0.12	3372.62	80.65	1093.03

4.3. The Results of Simulating the Architectural Model

To ensure the usability of our proposed blockchain-based PHR system in practice, the architectural model evaluates the proposed blockchain-based PHR system by simulating different arrival rates. The different arrival rate can represent the different numbers of population because multiplying the arrival rate with time may result in the population. The arrival rate is estimated as shown below.

For instance, Kut Chap district which is a part of Udon Thani province in the north-east part of Thailand has seven subdistricts. Kut Chap district can represent a regular district in Thailand. There are 94 villages under the seven subdistricts with a total population of approximately 55,000 [50]. If we assume that everybody access the proposed blockchain-based PHR system three times each day, there will be 165,000 accesses per day for the whole district. Then, the arrival rate will be

$$\frac{165{,}000}{24\text{ h}} = \frac{165{,}000}{3600 * 24} = \frac{165{,}000}{86{,}400} = 1.0 \text{ per second} \quad (1)$$

Again, if each person accesses the proposed blockchain-based PHR system six times per day, the arrival rate will be twice and the arrival rate will become 3.8 per second or 330,000 accesses per day. If all people lived in Kut Chap district access the proposed blockchain-based PHR system every 1-h each day, the arrival rate will be 15.2 per second. By varying the arrival rate of the scenarios, large population is simulated and the performance of the proposed blockchain-based PHR system is evaluated. We focus on the performance of the proposed blockchain-based PHR system as the main aspect with respect to the expected number of users. Thus, the usage profile is created with the workload which has been approximated as an open workload with the properties as follow:

- The workload is memoryless. This means the previous states of the workload are irrelevant to the current state.
- The various arrival rate is used to test for various the number of users.
- The file size of PHR data range between 128 KB and 128 MB.
- The emergency response time 8 min [51] is used as acceptable response time to support a pre-hospital care [52].

In simulating the architectural model, three arrival rates including 1.9, 3.8 and 15.2 are used. The result of overall usage for various arrival rates is shown in Figure 9.

Figure 9 shows the cumulative function diagram of the simulation and it can be seen that most of the operations response within

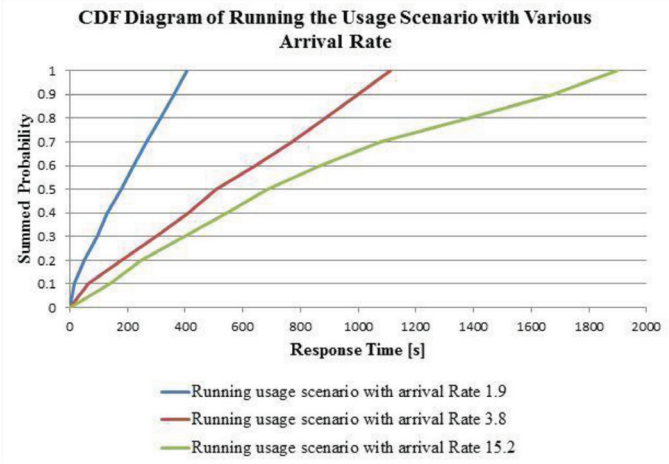


Figure 9 | Testing the overall model with various arrival rates

4 min when arrival rate is 1.9. It means that the overall system can serve around 165,000 operations per day with the service time within 4 min. Thus, the system can support three accesses per day for all people lived in Kut Chap district with the service time within 4 min. When the arrival rate is double (3.8), 50% of the operation response within 8 min and all the response times are <20 min. Thus, it can support 330,000 accesses per day for non-emergency cases. When the arrival rate is increased to 15.2 per second, 30% of the response times below 8 min. Thus, it can support 396,000 emergency cases per day even when the users access the proposed blockchain-based PHR system every hour.

5. CONCLUSION

In this paper, an architectural model is proposed with Palladio workbench to evaluate the blockchain-based PHR system previously proposed in Thwin and Vasupongayya [4]. A prototype system for our proposed blockchain-based PHR system is implemented to extract the key parameters for developing the architectural model. Then, the prototype system is experimented with the synthetic PHR data consisting of various data sizes including 128, 512 KB, 2, 8, 32 and 128 MB. The resulted execution times fall into two groups. In the first group, each execution time varies according to the PHR data size and the execution times. In the second group the execution time remains nearly the same. Thus, the execution times of the first group are modeled with a PDF while the execution times of the second group are modeled with their original values as a constant. To mimic the reality, the proposed architectural model is simulated with various arrival rates including 1.9, 3.8 and 15.2 per second respectively. These arrival rates can represent the large numbers of accesses including 165,000, 330,000 and 1,320,000 accesses each day. Kut Chap district has around 55,000 people and it can be a representative for a regular community in Thailand. If everybody accesses the proposed blockchain-based PHR system three times per day, the arrival rate is approximately 1.9 per second. The architectural model estimates that the proposed PHR systems can response within 4 min for the 165,000 accesses in each day. However, the result of simulating with the arrival rate of 3.8 per second shows that the response times for all operations is <20 min and 50% of those responses are within the 8 min of the emergency requirements. The result of simulating with the arrival rate of 15.2 per second shows that only 30% of the response times are within the emergency time of 8 min, however, using the PHR system every 1 h for all people in the district may be very rare case.

There are some limitations in this work. The hyperledger blockchain is executed with a dummy consensus. The network between each machine is modelled with no significant network delays. The population is assumed to be equally distributed. However, the proposed architectural models also provide a basis for future research into optimal system configuration such as non-functional properties. The future research plan includes further evaluating a distributed version of the proposed blockchain-based PHR system and also designing the tuning parameters for the proposed blockchain-based PHR system to adjust to the increasing in the population size and accessing patterns.

CONFLICTS OF INTEREST

The authors declare they have no conflicts of interest.

ACKNOWLEDGMENTS

We may express the thankful to our scholarship program. This research was supported by the Higher Education Research Promotion and the Thailand's Education Hub for Southern Region of ASEAN Countries Project Office of the Higher Education Commission. This work is the extension of our previous manuscript which is publishing in the open access journal, Security and Communication Networks.

REFERENCES

[1] P.C. Tang, J.S. Ash, D.W. Bates, J.M. Overhage, D.Z. Sands, Personal health records: definitions, benefits, and strategies for overcoming barriers to adoption, J. Am. Med. Inform. Assoc. 13 (2006), 121–126.

[2] J. Burrington-Brown, J. Fishel, L. Fox, B. Friedman, K. Giannangelo, E. Jacobs, et al. Defining the personal health record. AHIMA releases definition, attributes of consumer health record, J. AHIMA 76 (2005), 24–25.

[3] T.T. Thwin, S. Vasupongayya, Blockchain based secret-data sharing model for personal health record system, 2018 5th International Conference on Advanced Informatics: Concept Theory and Applications (ICAICTA), IEEE, Krabi, Thailand, 2018, pp. 196–201.

[4] T.T. Thwin, S. Vasupongayya, Blockchain-based access control model to preserve privacy for personal health record systems. Secur. Commun. Netw. 2019 (2019).

[5] A. Brunnert, A. van Hoorn, F Willnecker, A. Danciu, W. Hasselbring, C. Heger, et al. Performance-oriented DevOps: A Research Agenda, ArXiv150804752 Cs, 2015.

[6] S. Becker, H. Koziolek, R. Reussner, The Palladio component model for model-driven performance prediction, J. Syst. Softw. 82 (2009), 3–22.

[7] S. Becker, H. Koziolek, R. Reussner, Model-based performance prediction with the palladio component model, Proceedings of

the 6th International Workshop on Software and Performance (WOSP), ACM, New York, NY, USA, 2007, pp. 54–65.
[8] R.L. Coffield, J. Ishee, J.L. Kapp, K.D. Lyles, R.L. Williams, Jones Day – Personal Health Records: History, Evolution, and the Implications of the ARRA, American Health Lawyers Association Member Briefing, 2011, available from: http://www.jonesday.com/personal-health-records-history-evolution-and-the-implications-or-arra-iamerican-health-lawyers-associationi-01-01-2011/ [Online].
[9] Consumers and Health Information Technology: A National Survey - CHCF.org, available from: http://www.chcf.org/publications/2010/04/consumers-and-health-information-technology-a-national-survey [Online].
[10] S. Nakamoto, Bitcoin: a peer-to-peer electronic cash system, 2008, available from: https://bitcoin.org/bitcoin.pdf [Online].
[11] M. Li, S. Yu, Y. Zheng, K. Ren, W. Lou, Scalable and secure sharing of personal health records in cloud computing using attribute-based encryption, IEEE Trans. Parallel Distrib. Syst. 24 (2013), 131–143.
[12] M.H. Au, T.H. Yuen, J.K. Liu, W. Susilo, X. Huang, Y. Xiang, et al. A general framework for secure sharing of personal health records in cloud system, J. Comput. Syst. Sci. 90 (2017), 46–62.
[13] Y. Sreenivasa Rao, A secure and efficient Ciphertext-policy attribute-based signcryption for personal health records sharing in cloud computing, Future Gener. Comput. Syst. 67 (2017), 133–151.
[14] H.S. Gardiyawasam Pussewalage, V.A. Oleshchuk, A distributed multi-authority attribute based encryption scheme for secure sharing of personal health records, Proceedings of the 22nd ACM on Symposium on Access Control Models and Technologies, ACM, New York, NY, USA, 2017, pp. 255–262.
[15] Y. Liu, Y. Zhang, J. Ling, Z. Liu, Secure and fine-grained access control on e-healthcare records in mobile cloud computing, Future Gener. Comput. Syst. 78 (2018), 1020–1026.
[16] K. He, J. Weng, J.K. Liu, W. Zhou, J.N. Liu, Efficient fine-grained access control for secure personal health records in cloud computing, International Conference on Network and System Security (NSS), Springer, Cham, 2016, pp. 65–79.
[17] W.M. Li, X.L. Li, Q.Y. Wen, S. Zhang, H. Zhang, Flexible CP-ABE based access control on encrypted data for mobile users in hybrid cloud system. J. Comput. Sci. Technol. 32 (2017), 974–990.
[18] K. Gu, W. Jia, G. Wang, S. Wen, Efficient and secure attribute-based signature for monotone predicates, Acta Inform. 54 (2017), 521–541.
[19] D. Sangeetha, V. Vaidehi, A secure cloud based personal health record framework for a multi owner environment, Ann. Telecommun. 72 (2017), 95–104.
[20] A. Azaria, A. Ekblaw, T. Vieira, A. Lippman, MedRec: using blockchain for medical data access and permission management, 2016 2nd International Conference on Open and Big Data (OBD), IEEE, Vienna, Austria, 2016, pp. 25–30.
[21] L.A. Linn, M.B. Koo, Blockchain for health data and its potential use in health IT and health care related research, ONC/NIST Use of Blockchain for Healthcare and Research Workshop, Gaithersburg, MD, USA, 2016.
[22] D. Ivan, Moving toward a blockchain-based method for the secure storage of patient record, ONC/NIST Use of Blockchain for Healthcare and Research Workshop, Gaithersburg, MD, USA, 2016.
[23] H. Wang, Y. Song, Secure cloud-based EHR system using attribute-based cryptosystem and blockchain. J. Med. Syst. 42 (2018), 152–161.
[24] H. Li, L. Zhu, M. Shen, F. Gao, X. Tao, S. Liu, Blockchain-based data preservation system for medical data, J. Med. Syst. 42 (2018), 141.
[25] A. Roehrs, C.A. da Costa, R. da Rosa Righi, OmniPHR: a distributed architecture model to integrate personal health records, J. Biomed. Inform. 71 (2017), 70–81.
[26] A. Roehrs, C.A. da Costa, R. da Rosa Righi, V.F. da Silva, J.R. Goldim, D.C. Schmidt, Analyzing the performance of a blockchain-based personal health record implementation, J. Biomed. Inform. 92 (2019), 103140.
[27] T. de Gooijer, A. Jansen, H. Koziolek, A. Koziolek, An industrial case study of performance and cost design space exploration, Proceedings of the 3rd ACM/SPEC International Conference on Performance Engineering, ACM, New York, NY,USA, 2012, pp. 205–216.
[28] F. Willnecker, A. Brunnert, Predicting energy consumption by extending the palladio component model, Symposium on Software Performance (SOSP), 2014, pp. 177–188, available from: https://www.fortiss.org/veroeffentlichungen/publikationen/details/predicting-energy-consumption-by-extending-the-palladio-component-model [Online].
[29] A. Brunnert, C. Vögele, H. Krcmar, Automatic performance model generation for Java enterprise edition (EE) applications, European Workshop on Computer Performance Engineering (EPEW), Springer, Berlin, Heidelberg, 2013, pp. 74–88.
[30] S.S.M. Chow, J. Weng, Y. Yang, and R. H. Deng, Efficient unidirectional proxy re-encryption, Progress in Cryptology – AFRICACRYPT, Springer, Berlin, Heidelberg, 2010, pp. 316–332.
[31] G. Ateniese, K. Fu, M. Green, S. Hohenberger, Improved proxy re-encryption schemes with applications to secure distributed storage, ACM Trans. Inf. Syst. Secur. 9 (2006), 1–30.
[32] M.J. Dworkin, E.B. Barker, J.R. Nechvatal, J. Foti, L.E. Bassham, E. Roback, et al. Advanced Encryption Standard (AES), Federal Information Processing Standards Publication, 2001, available from: https://www.nist.gov/publications/advanced-encryption-standard-aes [Online].
[33] "Docker", Docker, available from: https://www.docker.com/ [Online].
[34] "npm – Orgs", available from: https://www.npmjs.com/ [Online].
[35] D. Sobhy, Y. El-Sonbaty, M.A. Elnasr, MedCloud: healthcare cloud computing system, 2012 International Conference for Internet Technology and Secured Transactions, IEEE, London, UK, 2012, pp. 161–166.
[36] M. Wangthammang, S. Vasupongayya, Distributed storage design for encrypted personal health record data, 2016 8th International Conference on Knowledge and Smart Technology (KST), IEEE, Chiangmai, Thailand, 2016, pp. 184–189.
[37] W. Lijun, H. Yongfeng, C. Ji, Z. Ke, L. Chunhua, Medoop: a medical information platform based on Hadoop, 2013 IEEE 15th International Conference on e-Health Networking, Applications and Services (Healthcom), IEEE, Lisbon, Portugal, 2013, pp. 1–6.
[38] Y. Li, L. Guo, Y. Guo, An efficient and performance-aware big data storage system, International Conference on Cloud Computing and Services Science (CLOSER), Springer, Cham, 2012, pp. 102–116.
[39] A. Bahga, V.K. Madisetti, A cloud-based approach for interoperable electronic health records (EHRs), IEEE J. Biomed. Health Inform. 17 (2013), 894–906.
[40] Y. Li, L. Guo, C. Wu, C. Lee, Y. Guo, Building a cloud-based platform for personal health sensor data management, IEEE-EMBS

International Conference on Biomedical and Health Informatics (BHI), IEEE, Valencia, Spain, 2014, pp. 223–226.
[41] B. Dong, Q. Zheng, F. Tian, K.M. Chao, R. Ma, R. Anane, An optimized approach for storing and accessing small files on cloud storage, J. Netw. Comput. Appl. 35 (2012), 1847–1862.
[42] "YouTube", available from: https://www.youtube.com/ [Online].
[43] "VA Personal Health Record Sample Data - Data.gov", available from: https://catalog.data.gov/dataset/va-personal-health-record-non-identifiable-data [Online].
[44] "MedCram - Best Medical Lectures and Medical Videos, CME, CE", available from: https://www.medcram.com [Online].
[45] "MP4Tools - Home", available from: https://www.mp4joiner.org/en/ [Online].
[46] "Machete Video Editor Lite - Free Video Editor", available from: http://www.machetesoft.com/about-machete-video-editor-lite.html [Online].
[47] "Freetime software", available from: http://www.pcfreetime.com/ [Online].
[48] "Avidemux - Main Page", available from: http://avidemux.sourceforge.net/ [Online].
[49] "FREE Video Converter by Freemake: Convert MP4 MP3 250 File Formats" Freemake.
[50] "Kut Chap (District, Udon Thani, Thailand) - Population Statistics, Charts, Map and Location", available from: https://www.citypopulation.de/php/thailand-admin.php?adm2id=4102 [Online].
[51] P. Thummavet, S. Vasupongayya, Privacy-preserving emergency access control for personal health records, Maejo Int. J. Sci. Technol. 9 (2015), 108–120.
[52] S. Hasavari, Y.T. Song, A secure and scalable data source for emergency medical care using blockchain technology, Int. J. Netw. Distrib. Comput. 7 (2019), 158–166.